IMPROVED FINITE-ELEMENT METHOD FOR SOLUTE TRANSPORT

By Fang Xin Yu[1] and Vijay P. Singh,[2] Member, ASCE

ABSTRACT: Five major modifications to the Galerkin finite-element formulation for solute transport were made in this study: (1) A mixed formulation for the time-derivative term of the governing equation was developed by combining the Galerkin method and the collocation method; (2) a general and useful formulation for the advection and dispersion terms was derived by applying Green's theorem so that any given advection-dominated boundary conditions can be correctly handled; (3) simpler expressions for leaky boundary conditions and surface flux conditions were developed using the unit step function; (4) nonambiguous expressions of the source and sink terms were derived using the Dirac delta function; and (5) a finite-integration solution scheme was developed to solve the system of ordinary differential equations, and a discussion critical to the use of the finite-difference solution scheme was presented. The effects of these five modifications on numerical solution were investigated.

INTRODUCTION

One of the major differences between ground-water flow modeling and solute-transport modeling is the existence of the advective term in the governing equation of solute transport. This term makes the system of equations nonsymmetric and prone to severe numerical oscillation, dispersion, or instability, when obtaining its numerial solution. A careful check of the conventional Galerkin formulation reveals that the continuity of solute flux through a generic-element boundary surface will fail for the prescribed advection-dominated boundaries. This is because the conventional finite-element formulation or its variation was derived without applying Green's theorem to the advective term of the governing partial-differential equation (PDE) (Istok 1989; Huyakorn et al. 1986; Bear and Verruijt 1987). Consequently, only the dispersive flux, instead of the total flux (sum of the advective and dispersive fluxes), could be specified at the boundaries with observations. Although Pinder and Gray (1977) pointed out this problem as early as 1977 and proposed the formula for two-dimensional mass transport, many subsequent researchers and writers did not pay much attention to it (Wang and Anderson 1982; Segerlind 1984; Bear and Verruijt 1987, Istok 1989; Thomson et al. 1984). Huyakorn et al. (1986) developed a three-dimensional (3D) finite-element model to simulate multiple-layered aquifer systems. However, their model, cannot handle advection-dominated boundary conditions. The weakness of the model is probably not caused by the complexity of additional programming, rather, it is caused by using the incorrect numerical formulation for the advection term of the governing PDE.

During the application of the finite-element method (FEM), numerical oscillation, dispersion, and instability often occur. Some of the major factors causing these problems are: (1) improper handling of the time-derivative term of the governing equation; (2) the improper choice of a time step size; (3) the improper choice of a solution scheme to solve the system of ordinary-differential equations; and (4) the improper choice of element size and shape. Currently, there are two types of formulations in handling the time-derivative term—one being the consistent formulation, and the other the lumped formulation (Istok 1989). The consistent formulation uses the same function form to represent both the basis functions and the weight functions for both the time- and space-derivative terms, whereas the lumped formulation defines a set of special weight and basis functions to handle the time-derivative term. Although the lumped formulation is inconsistent in terms of the overall formulation, it is less susceptible to numerical oscillation.

To overcome some of the numerical problems of the Galerkin FEM for advection-dominated flow, Sun and Yeh (1983) proposed a two-dimensional (2D) upstream weight multiple-cell balance model. Later, Wang et al. (1986) extended it to 3D flows. Although their model showed an improvement over the conventional Galerkin FEM, numerical problems, such as numerical oscillation, are still experienced for large Peclet numbers (Yeh 1986). To further enhance the model's capability in handling numerical problems, Yeh (1986) modified the Wang et al. (1986) model by requiring the set of weight functions to be orthogonal to the set of basis functions. As a result, Yeh's formulation is more powerful in handling numerical problems, but is also

[1]Res. Assoc., Louisiana Transp. Res. Ctr., 4101 Gourrier Ave., Baton Rouge, LA 70808.
[2]Prof., Dept. of Civ. Engrg., Louisiana State Univ., Baton Rouge, LA 70803.

Note. Discussion open until July 1, 1995. To extend the closing date one month, a written request must be filed with the ASCE Manager of Journals. The manuscript for this paper was submitted for review and possible publication on September 23, 1992. This paper is part of the ***Journal of Hydraulic Engineering***, Vol. 121, No. 2, February, 1995. ©ASCE, ISSN 0733-9429/95/0002-0145–0158/$2.00 + $.25 per page. Paper No. 4845.

more complicated to program. Likewise, Park and Liggett (1990, 1991) developed a Taylor-least-square finite-element method. Hermite cubic functions were used for element shape representation. The method was demonstrated to be effective for advection-dominated flow. However, the method also requires more computer space and time to run the model.

The finite-difference solution scheme (FDSS) is most frequently used to solve the final system of ordinary-differential equations. The limitations of using the FDSS have not been thoroughly investigated. The weight factor in the FDSS may be subjectively chosen between zero and one, and is often chosen as one because of its unconditional stability property. The results obtained in this study show the unity choice cannot yield a theoretically exact solution in most practical applications.

The objective of this study is to modify the Galerkin numerical formulation for solute transport in a ground-water flow system. The modification mainly consisted of six parts: (1) To propose a consistent way to derive the lumped formulation; (2) to derive a general numerical formulation for advective and dispersive terms so that prescribed advection-dominated boundaries can be handled correctly; (3) to derive a simpler and more practical expression for leaky boundary conditions; (4) to derive proper expressions for both sources and sinks; (5) to develop a finite-integration solution scheme (FISS) to solve the system of ordinary-differential equations; and (6) to investigate effects of the five numerical modifications on numerical solutions.

GOVERNING EQUATION

The governing equation for solute transport can be derived by using the mass balance principle, along with certain assumptions [Bear (1979); Istok (1989); Yu (1992)] as

$$\frac{\partial(R\theta C)}{\partial t} = \nabla\cdot[D_m]\nabla(\theta C) - \nabla\cdot(C\{q\}) - \lambda\theta RC + \sum_{i=1}^{n_w} Q_i(t)C\delta(x - x_i, y - y_i, z - z_i) + \sum_{i=1}^{n_s} Q_i(t)C_i^*\delta(x - x_i, y - y_i, z - z_i) \quad (1)$$

where C = solute concentration, (ML^{-3}); n_s = number of sources in study domain; n_w = number of pumping wells or sinks in domain; θ = medium porosity; λ = first-order biochemical reaction coefficient, (T^{-1}); C_i^* = solute concentration at the ith source point, (ML^{-3}); $Q_i(t)$ = source rate of fluid at ith source point, (L^3T^{-1}); $Q_j(t)$ = sink rate of fluid, (L^3T^{-1}), negative in value at the jth sink point; ρ_b = bulk density of porous media, (ML^{-3}); K_d = distribution coefficient, (L^3M^{-1}); $\{q\}$ = apparent velocity vector; $[D_m]$ = combined mechanical and molecular diffusion coefficient tensor; and R = retardation factor defined as $R = 1 + \rho_b K_d/\theta$. It is important to note that solute concentration for the pumped water is unknown, whereas solute concentration in the recharged water is normally known. In many of the past studies, they have not been properly treated (Istok 1989).

INITIAL AND BOUNDARY CONDITIONS

The initial condition can generally be expressed as

$$C = g_1(x, y, z, 0); \qquad (x, y, z) \in D \quad (2)$$

where g_1 = a known positive function over the study domain D. Some of the practical boundary conditions are described as follows: (1) Boundary of prescribed concentration (type 1)

$$C = g_2(x, y, z, t); \qquad (x, y, z) \in S;\ t \geq 0 \quad (3)$$

where g_2 = a known positive function over the boundary surface S at time t; (2) boundary of prescribed flux (type 2)

$$(\{q\}C - \theta[D_m]\nabla C)^T\cdot\{n'\} = -g_3(x, y, z, t);\ (x, y, z) \in S \quad (4)$$

where g_3 = a known function (positive for inflow) over the boundary surface S and has the units (MT^{-1}/L^2); and $\{n'\}$ = unit outward vector normal to the boundary surface S. A special case of (4) is the impervious boundary

$$(\{q\}C - \theta[D_m]\nabla C)^T\cdot\{n'\} = 0; \qquad (x, y, z) \in S,\ t \geq 0 \quad (5)$$

(3) leaky boundary (type 3), which can be better illustrated by a system of two aquifers separated by a thin layer of aquitard. On top of the aquitard is a water-table aquifer and below the aquitard is a confined aquifer, which is the study domain. Let C and h be the solute concentration and piezometric head at the boundary between the confined aquifer and the aquitard, and C^* and h_0 be the solute concentration and the hydraulic head at the boundary between the watertable aquifer and the aquitard. By assuming that storage in the aquitard is negligible and that Fick's law and Darcy's law are valid for the flow, the total solute flux can be expressed by

$$(\{q\}C - \theta[D_m]\nabla C)^T \cdot \{n^o\} = D_1 \frac{\theta_1(C - C^*)}{B_1} + k_1 \frac{h - h_0}{B_1}(u(h - h_0)C + u(h_0 - h)C^*) \quad (6)$$

Eq. (6) may be written in a more convenient form as

$$(\{q\}C - \theta[D_m]\nabla C)^T \cdot \{n^o\} = (c_2 + c_1(h - h_0)u(h - h_0))C - (c_2 - c_1(h - h_0)u(h_0 - h))C^* \quad (7)$$

where θ = porosity of confined aquifer; and θ_1 = porosity of aquitard, $c_1 = k_1/B_1$ and $c_2 = \theta_1 D_1/B_1$ = two aquifer parameters, in which B_1 = thickness of aquitard, k_1 = hydraulic conductivity coefficient, and D_1 = dispersivity coefficient of aquitard. The function $u(\ )$ in (7) = unit step function defined by

$$u(\xi) = \begin{cases} 0 & \xi < 0 \\ 1 & \xi \geq 0 \end{cases} \quad (8)$$

The leaky boundary condition in (8) can be applied to any leaky boundaries such as an aquifer separated by an aquitard from a reservoir at one boundary of the aquifer.

NUMERICAL FORMULATION

For purposes of generalization, let the governing equation have the general form

$$L[\phi(x, y, z, t)] - F(x, y, z, t) = 0 \quad (9)$$

where L = a differential operator; ϕ = field variable; and F = some known function. Since the interpolation function is usually defined in an element-wise fashion (Istok 1989), an approximate solution $\Phi(x, y, z, t)$ over the entire domain can be expressed by

$$\hat{\Phi}(x, y, z, t) = \sum_{i=1}^{m} \hat{\phi}_i(x, y, z, t) = \sum_{i=1}^{m} \left[\sum_{j=1}^{n} N_j(x, y, z)\phi_j(t) \right]_i \quad (10)$$

where $\hat{\phi}_i(x, y, z, t)$ = interpolation function for the ith element; m = number of elements in the mesh; n = number of nodes of an element e; N_j = jth nodal basis function of an element e; and ϕ_j = value of the field variable for the jth node at time t. When the approximate solution of (10) is substituted into the governing (9) and is required to satisfy certain initial and boundary conditions, an error may exist at each point of the solution domain. If the principle of weighted residuals is applied, the sum of the weighted residuals at each node is forced to zero

$$\int\int_D\int w_i(x, y, z)R(x, y, z, t)\, dx\, dy\, dz$$

$$= \int\int_D\int w_i(x, y, z)\{L[\hat{\Phi}(x, y, z, t)] - F(x, y, z, t)\}\, dx\, dy\, dz = 0 \quad (11)$$

where $w_i(x, y, z)$ = weight function for the ith nodal point. In the past the Galerkin weight function was usually defined as

$$w_i(x, y, z) = N_i(x, y, z) \quad (12)$$

It is not clear as to how it can be used in the overall formulation such as (11). For consistence, the Galerkin weight function should be defined as

$$w_i(x, y, z) = \sum_{e=1}^{n_i} N_i^{(e)}(x, y, z); \qquad i = 1, 2, \ldots, p \quad (13)$$

where n_i = number of elements attached to the node i; and p = number of nodes in the mesh. By this definition, a clearer numerical derivation of the FEM could be presented. By substituting (13) into (11) one gets

$$R_i = \int\int_{D_i}\int \left\{ \sum_{e=1}^{n_i} N_i^{(e)}(x, y, z) \right\} \{L[\hat{\Phi}(x, y, z, t)] - F(x, y, z, t)\}\, dV \quad (14a)$$

$$R_i = \sum_{e=1}^{n_i} \int\int_{D_e}\int N_i^{(e)}(x, y, z)\{L[\hat{\phi}(x, y, z, t)]^{(e)} - F(x, y, z, t)\}\, dV \quad (14b)$$

$$R_i = \sum_{e=1}^{n_i} R_i^{(e)} = 0; \qquad i = 1, 2, \ldots, p \quad (14c)$$

where D_i = subdomain that consists of all elements attached to node i; $dV = dx\, dy\, dz$; D_e = element domain; $\hat{\phi}^{(e)}$ = element interpolation function; $R_i^{(e)}$ = weighted residual at node i contributed by an element e; $N_i^{(e)}$ = basis function for node i of an element e; and R_i = sum

of weighted residuals at node i. Thus, (14) produces a system of p equations. By solving this system of equations, one gets the values of the field variable at these p nodes. From (14), the residual at the ith node contributed by an element e, $R_i^{(e)}$, can be written as

$$R_i^{(e)} = -\int\int_{D_e}\int N_i(x, y, z)\{L[\hat{\phi}(x, y, z, t)] - F(x, y, z, t)\}\, dV;$$

$$i = 1, 2, \ldots, n; \qquad e = 1, 2, \ldots, p \tag{15}$$

The negative sign in (15) is arbitrarily added for later convenience. The residual vector for an element with n nodes can be expressed as

$$\{R^{(e)}\} = \begin{Bmatrix} R_1^{(e)} \\ \vdots \\ R_n^{(e)} \end{Bmatrix} = -\int\int_{D_e}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \{L[\hat{\phi}(x, y, z, t)] - F(x, y, z, t)\}\, dV \tag{16}$$

To apply (16) to solute transport in particular, let us rewrite (1) as

$$R\frac{\partial C}{\partial t} = \nabla\cdot[D_m]\nabla C - \nabla\cdot\left(\frac{\{q\}C}{\theta}\right) - \lambda RC + \frac{1}{\theta}\sum_{j=1}^{n_w} Q_j(t)C\delta(x - x_j, y - y_j, z - z_j)$$

$$+ \frac{1}{\theta}\sum_{i=1}^{n_s} Q_i(t)C_i^*\delta(x - x_i, y - y_i, z - z_i) \tag{17}$$

By comparing (17) with (9) and adopting the following generalized element interpolation function

$$\hat{\phi}(x, y, z, t) = \hat{C}(x, y, z, t) = \sum_{i=1}^{n} N_i(x, y, z)C_i(t) \tag{18}$$

(16) becomes

$$\{R^{(e)}\} = -\int\int_{D}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\frac{1}{\theta}\sum_{k=1}^{n_s} Q_kC_k^*\delta(x - x_k, y - y_k, z - z_k)\right) dV$$

$$- \int\int_{D}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\frac{1}{\theta}\sum_{j=1}^{n_w} Q_j\hat{C}\delta(x - x_j, y - y_j, z - z_j)\right) dV$$

$$- \int\int_{D}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\nabla\cdot[D_m]\nabla\hat{C} - \nabla\cdot\frac{\{q\}\hat{C}}{\theta} - \lambda R\hat{C} - R\frac{\partial\hat{C}}{\partial t}\right) dV \tag{19}$$

Now let us integrate (19) term by term as follows: First, handling the source and sink terms by using the integration property of the delta function and the general properties of a selected basis function, one gets (Yu 1992)

$$-\int\int_{D}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\frac{1}{\theta}\sum_{k=1}^{n_s} Q_kC_k^*\delta(x - x_k, y - y_k, z - z_k)\right) dV$$

$$- \int\int_{D}\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\frac{1}{\theta}\sum_{j=1}^{n_w} Q_j\hat{C}\delta(x - x_j, y - y_j, z - z_j)\right) dV = -\begin{Bmatrix} \frac{Q_1C_1^*}{\theta} \\ \vdots \\ \frac{Q_nC_n^*}{\theta} \end{Bmatrix} - [D_q]\begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} \tag{20}$$

where $[D_q]$ = element matrix contributed by all sink points within an element e

$$[D_q] = \frac{1}{\theta}\sum_{j=1}^{n_w} Q_j(t)\begin{bmatrix} N_1 & 0 & \cdots & 0 \\ 0 & N_2 & \cdots & 0 \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ \cdot & \cdot & \cdots & \cdot \\ 0 & 0 & \cdots & N_n \end{bmatrix}_{(x_j, y_j, z_j)} \tag{21}$$

Second, by handling the time derivative term, which can generally be expressed by

$$\int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [R] \frac{\partial \hat{C}}{\partial t}\, dx\, dy\, dz = [A] \begin{Bmatrix} \frac{\partial C_1}{\partial t} \\ \vdots \\ \frac{\partial C_n}{\partial t} \end{Bmatrix} \tag{22}$$

where $[A]$ = element sorption matrix. There can be many ways to compute the sorption matrix depending on how one defines the weight function and interpolation function for the time derivative. Two of the most frequently used formulations are the consistent formulation and lumped formulation. The consistent formulation uses the Galerkin-basis functions to represent both weight functions and interpolation functions, for both spatial-derivative terms and the time-derivative term and is written as

$$[A] = R \int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [N_1 \ldots N_n]\, dx\, dy\, dz \tag{23}$$

The lumped formulation defines a set of special functions to represent the weight functions and interpolation functions (Yu 1992). As a result, the sorption matrix for the lumped formulation is presented as

$$[A] = \frac{R}{n} [I]_{nXn} \int\int_D\int dx\, dy\, dz \tag{24}$$

where $[I]$ = identity matrix. The lumped formulation changes the Galerkin weight function to a specially defined function, and causes the overall numerical formulation to be inconsistent. One way to overcome this theoretical deficiency is to keep the Galerkin weight function the same in the overall formulation, but to interpolate the time derivative by the collocation method. This leads to the Galerkin-collocation or mixed formulation

$$[A] = \int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [R][\delta_1(x - x_1, y - y_1, z - z_1) \ldots \delta_n(x - x_n, y - y_n, z - z_n)]\, dV \tag{25}$$

It can be shown that (25) will reduce to the same expression as that of the lumped formulation of (24). In this way, however, the overall numerical formulation is consistent except that the time-derivative term is interpolated differently.

Third, by handling the first-order biochemical reaction term

$$\int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [\lambda R]\hat{C}\, dx\, dy\, dz = [D_f] \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} = \left(\int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [\lambda R][N_1 \ldots N_n]\, dx\, dy\, dz \right) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} \tag{26}$$

where $[D_f]$ = a matrix contributed by the first-order biochemical reaction and the sorption/desorption process; (4) by handling the advection and dispersion terms; both of which, in (19), can be written as

$$\int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left(\nabla \cdot \frac{\{q\}\hat{C}}{\theta} - \nabla \cdot [D_m] \nabla \hat{C} \right) dx\, dy\, dz = \int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \nabla^2 \Psi\, dx\, dy\, dz \tag{27}$$

where

$$\nabla \Psi = \frac{\{q\}\hat{C}}{\theta} - [D_m] \nabla \hat{C} \tag{28}$$

Let us recall the divergence theorem (Wylie and Barrett 1982). By defining a vector $\{F\} = u\nabla v$, in which u and v are two scalar variables, the divergence theorem can be written as

$$\int\int_D\int u \nabla^2 v\, dV = \oint_S u \nabla v \cdot \{n^o\}\, ds - \int\int_D\int \nabla u \cdot \nabla v\, dV \tag{29}$$

By applying the divergence theorem of (29) to (27), one gets

$$\int\int_D\int \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \nabla^2 \Psi\, dx\, dy\, dz = \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta} \right] (\{q\}\hat{C} - \theta [D_m] \nabla \hat{C}) \cdot \{n^o\}\, ds$$

J. Hydraul. Eng., 1995, 121(2): 145-158

$$-\int\int_D\int \nabla \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \cdot \left(\frac{\{q\}\hat{C}}{\theta} - [D_m]\nabla\hat{C} \right) dx\, dy\, dz \quad (30)$$

By using the interpolation function of (18), the integration over the element domain D in (30) can be expressed as

$$-\int\int_D\int \nabla \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \cdot \left(\frac{\{q\}\hat{C}}{\theta} - [D_m]\nabla\hat{C} \right) dx\, dy\, dz = ([D_d] - [D_a]) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} \quad (31)$$

where $[D_m]$ = dispersion and diffusion tensor and can be computed by equations given by Wang et al. (1986); and $[D_d]$ and $[D_a]$ = element dispersion and advection matrices, respectively, and are defined as

$$[D_d] = \int\int_D\int \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial N_n}{\partial x} & \frac{\partial N_n}{\partial y} & \frac{\partial N_n}{\partial z} \end{bmatrix} [D_m] \begin{bmatrix} \frac{\partial N_1}{\partial x} & \cdots & \frac{\partial N_n}{\partial x} \\ \frac{\partial N_1}{\partial y} & \cdots & \frac{\partial N_n}{\partial y} \\ \frac{\partial N_1}{\partial z} & \cdots & \frac{\partial N_n}{\partial z} \end{bmatrix} dV \quad (32)$$

$$[D_a] = \int\int_D\int \begin{bmatrix} \frac{\partial N_1}{\partial x} & \frac{\partial N_1}{\partial y} & \frac{\partial N_1}{\partial z} \\ \vdots & \vdots & \vdots \\ \frac{\partial N_n}{\partial x} & \frac{\partial N_n}{\partial y} & \frac{\partial N_n}{\partial z} \end{bmatrix} \begin{Bmatrix} \frac{q_x}{\theta} \\ \frac{q_y}{\theta} \\ \frac{q_z}{\theta} \end{Bmatrix} [N_1 \ldots N_n]\, dV \quad (33)$$

Substituting the preceding element integration expressions into (19) yields

$$\{R^{(c)}\} = ([D_d] - [D_a] + [D_f] - [D_a]) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} + [A] \begin{Bmatrix} \frac{\partial C_1}{\partial t} \\ \vdots \\ \frac{\partial C_n}{\partial t} \end{Bmatrix} - \begin{Bmatrix} \frac{Q_1 C_1^*}{\theta} \\ \vdots \\ \frac{Q_n C_n^*}{\theta} \end{Bmatrix} + \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \begin{bmatrix} 1 \\ \theta \end{bmatrix} (\{q\}\hat{C} - \theta[D_m]\nabla\hat{C}) \cdot \{n^o\}\, ds \quad (34)$$

It is important to note that (34) is different from the conventional Galerkin formulation (Wang and Anderson 1982; Huyakorn et al. 1986; Istok 1989; Segerlind 1984; Bear and Verruijt 1987). In the formulations reported in the literature, the advection term is usually integrated directly without using the divergence theorem. As a result, the boundary surface integral consists of only the dispersion flux term. To clearly show the difference between the modified formulation and conventional Galerkin finite-element formulation, the conventional Galerkin formulation may be written as

$$\{R^{(c)}\} = ([D_d] + [D_a] + [D_f]) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} + [A] \begin{Bmatrix} \frac{\partial C_1}{\partial t} \\ \vdots \\ \frac{\partial C_n}{\partial t} \end{Bmatrix} - \begin{Bmatrix} \frac{Q_1 C_1^*}{\theta} \\ \vdots \\ \frac{Q_n C_n^*}{\theta} \end{Bmatrix} - \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} ([D_m]\nabla\hat{C}) \cdot \{n^o\}\, ds \quad (35)$$

It is clear that when the conventional Galerkin formulation is used, only the dispersion flux condition can be specified. In some practical problems, however, the total flux may be observed and used as input. In such cases, any attempt to specify the observed fluxes will lead to the violation of the mass-conservation law. On the other hand, the modified formulation can handle practically any type of boundary conditions. This is because the surface integral in (34) deals with both the advection and dispersion fluxes. The sum of element surface integrals for all interior element surfaces are conventionally assumed to be zero. Therefore, the surface integral needs to be considered only for those element surfaces that coincide with the solution boundaries. For a boundary of prescribed concentration, the known concentration values at the boundary

will be directly assigned to their corresponding nodes. For an impervious boundary, the surface integral in (34) is zero. For a boundary with prescribed flux, the surface integral in (34) can be expressed, using (4), as

$$\oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta}\right] (\{q\}\hat{C} - \theta[D_m]\nabla\hat{C})\cdot\{n_o\}\, ds = -\oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{q_3}{\theta}\right] ds \tag{36}$$

For a leaky boundary, the surface integral in (34) can be expressed as

$$\oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta}\right] (\{q\}\hat{C} - \theta[D_m]\nabla\hat{C})\cdot\{n^o\}\, ds = \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta}\right] (c_2 + c_1(\bar{h} - \bar{h}_0)u(\bar{h} - \bar{h}_0))\hat{C}\, ds$$

$$- \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta}\right] (c_2 - c_1(\bar{h} - \bar{h}_0)u(\bar{h}_0 - \bar{h}))C^*\, ds \tag{37}$$

where $\bar{h}$ and $\bar{h}_0$ = averaged values of h and h_0, respectively, over the corresponding element surface. By using the element interpolation function of (18), the first integral on the right-hand side of (37) can be written for a generic element as

$$\oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{c_2 + c_1(\bar{h} - \bar{h}_0)u(\bar{h} - \bar{h}_0)}{\theta}\right] \hat{C}\, ds = \left(\sum_{i=1}^{n_k} [D_L]^{(S_i)}\right) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} \tag{38}$$

$$[D_L]^{(S_i)} = \left(\frac{c_2 + c_1(\bar{h} - \bar{h}_0)u(\bar{h} - \bar{h}_0)}{\theta}\right)^{(S_i)} \int\int_{S_i} \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} [N_1 \ldots N_n]\, ds; \qquad i = 1, 2, \ldots, n_k \tag{39}$$

where n_k = number of surfaces that an element has. For later convenience, let us define a surface constant $a(S_i)$

$$a(S_i) = \left(\frac{\bar{g}_3 + [c_2 - c_1(\bar{h} - \bar{h}_0)u(\bar{h}_0 - \bar{h})]C^*}{\theta}\right)^{(S_i)}; \qquad i = 1, 2, \ldots, n_k \tag{40}$$

The second surface integral on the right-hand side of (37) can be combined with (36) into one type of surface integral

$$-\oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{g_3}{\theta}\right] ds - \oint_S \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} \left[\frac{1}{\theta}\right] (c_2 - c_1(\bar{h} - \bar{h}_0)u(\bar{h}_0 - \bar{h}))C^*\, ds = -\sum_{i=1}^{n_k} a(S_i) \int\int_{S_i} \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} ds \tag{41}$$

By using the isoparametric scheme (Istok 1989) and the coordinate transformation scheme between the global coordinates and the natural coordinates, by means of the Jacobian matrix, and then applying the Gauss quadrature method to the preceding integrals, all element integrals can be carried out numerically (Istok 1989; Yu 1992). Finally, by substituting all the numerical expressions for various element matrices and vector integrals into the element residual vector (34), one gets

$$\{R^{(e)}\} = ([D_d] - [D_a] + [D_f] - [D_q] + [D_L]) \begin{Bmatrix} C_1 \\ \vdots \\ C_n \end{Bmatrix} + [A] \begin{Bmatrix} \frac{\partial C_1}{\partial t} \\ \vdots \\ \frac{\partial C_n}{\partial t} \end{Bmatrix} - \begin{Bmatrix} \frac{Q_1 C_1^*}{\theta} \\ \vdots \\ \frac{Q_n C_n^*}{\theta} \end{Bmatrix} - \begin{Bmatrix} f_1 \\ \vdots \\ f_n \end{Bmatrix} \tag{42}$$

where

$$\begin{Bmatrix} f_1 \\ \vdots \\ f_n \end{Bmatrix} = \sum_{i=1}^{n_k} a(S_i) \int\int_{S_i} \begin{Bmatrix} N_1 \\ \vdots \\ N_n \end{Bmatrix} ds; \qquad [D_L] = \sum_{i=1}^{n_k} [D_L]^{(S_i)} \tag{43}$$

Thus, (42) represents the general element residual vector. Once all the element vectors are computed, the corresponding element matrices and vectors can be expanded and assembled into the global matrices and vectors. By forcing the sum of the weighted residuals at each global node to be zero, one can obtain the global system of ordinary equations as

$$\{R\}_{\text{global}} = \sum_{e=1}^{m} \{R_e\}_{\text{expanded}} = \begin{Bmatrix} R_1 \\ \vdots \\ R_p \end{Bmatrix} = [D]\{C\} + [A]\{\dot{C}\} - \{F\} = \{0\} \tag{44}$$

where

$$[D] = \sum_{e=1}^{m} ([D_d] - [D_a] + [D_f] - [D_q] + [D_L])_{\text{expanded}} \tag{45}$$

$$\{C\} = \begin{Bmatrix} C_1 \\ C_2 \\ \vdots \\ C_p \end{Bmatrix}; \qquad \{\dot{C}\} = \begin{Bmatrix} \dfrac{\partial C_1}{\partial t} \\ \vdots \\ \dfrac{\partial C_p}{\partial t} \end{Bmatrix}; \qquad \{F\} = \sum_{e=1}^{m} \left[\begin{Bmatrix} \dfrac{Q_1 C_1^*}{\theta} \\ \vdots \\ \dfrac{Q_p C_p^*}{\theta} \end{Bmatrix} + \begin{Bmatrix} f_1 \\ \vdots \\ f_p \end{Bmatrix} \right] \tag{46a-c}$$

m = number of element in a finite-element mesh; and p = number of nodes in the mesh. Eq. (44) can be simply written as

$$[A]\{\dot{C}\} + [D]\{C\} = \{F\} \tag{47}$$

Eq. (47) is a system of ordinary-differential equations and is often solved by using the finite-difference solution scheme. However, under certain assumptions (to be discussed in the following section), (47) may be integrated from t to $t + \Delta t$ by applying the mean-value theorem as

$$[A](\{C\}_{t+\Delta t} - \{C\}_t) + [D]\{C(\xi_1)\}\Delta t = \{F(\xi_2)\}\Delta t; \qquad t \le \xi_1, \xi_2 \le t + \Delta t \tag{48}$$

where ξ_1 and ξ_2 = two unknown variables whose values lie in the time interval $(t, t + \Delta t)$. For purposes of comparison with the FDSS, we may use two weight factors w_1 and w_2 to replace ξ_1 and ξ_2, and (48) can be equivalently written as

$$[A](\{C\}_{t+\Delta t} - \{C\}_t) + [D](w_1\{C\}_{t+\Delta t} + (1 - w_1)\{C\}_t)\Delta t = (w_2\{F\}_{t+\Delta t} + (1 - w_2)\{F\}_t)\Delta t, \qquad 0 \le w_1, \quad w_2 \le 1 \tag{49}$$

By rearranging terms, we can write the FISS as

$$([A] + w_1\Delta t[D])\{C\}_{t+\Delta t} = ([A] - (1 - w_1)\Delta t[D])\{C\}_t + \Delta t(w_2\{F\}_{t+\Delta t} + (1 - w_2)\{F\}_t); \qquad 0 \le w_1 \le 1, 0 \le w_2 \le 1 \tag{50}$$

Two questions arise with regard to the solution scheme. First, under what assumptions can (47) be integrated using only two independent variables ξ_1 and ξ_2? Because $\{C\}$ and $\{F\}$ in (47) are vectors with p components, this is possible only in two special cases. Case 1: if the solute concentration and the sources at all the nodes increase or decrease linearly from time t to $t + \Delta t$, then $\{C(\xi)\} = 0.5(\{C(t)\} + \{C(t + \Delta t)\})$ and $\{F(\xi)\} = 0.5(\{F(t)\} + \{F(t + \Delta t)\})$; thus, the weight factors $w_1 = w_2 = 0.5$. Case 2: if the solute concentration vector $\{C\}$ and the source vector $\{F\}$ increase or decrease steadily during a time interval $[t, t + \Delta t]$ and can be expressed or approximated by the same type of power function of order n [e.g., $C_i(\tau) = C_i(t) \pm \lambda\tau^n$, $0 \le \tau \le \Delta t$, in which λ and n are two constants during time interval $[t, t + \Delta t]$, (47) can be integrated with $w_1 = n/(n + 1)$ and $w_2 = 1/(n + 1)$. Hence, w_1 and w_2 can theoretically be any value between zero and one. In reality, however, case 2 has little practical value because only under very special circumstances could $\{C\}$ and $\{F\}$ be increasing or decreasing steadily for all nodes, during $[t, t + \Delta t]$, and expressed by the same type of power function of the same order n.

Second, under what conditions can the FDSS become an exact solution scheme? Notice that when w_1 and $w_2 = w$, (50) becomes the FDSS. It follows that if and only if $w = 0.5$ can the FDSS become an exact solution scheme to (47), provided that solute concentration and the sources at any nodal point vary linearly during the time interval $[t, t + \Delta t]$. From this point of view, it is more appropriate to adjust the time step size Δt, rather than the weight factor w, in any numerical solute-transport modeling. Although the solute concentration distribution can sometimes be simulated quite accurately within a subdomain of a study, within which $\{C\}$ and $\{F\}$ increase or decrease monotonically, model verification based on these simple cases may be misleading and cannot be used in more complicated applications. In the past, the value of the weight factor in the FDSS is often subjectively chosen within [0, 1]. Many modelers, however, tend to use $w = 1$ due to its unconditional stability property. This study shows that $w = 1$ cannot provide an exact solution for complicated subsurface flows.

INVESTIGATION OF EFFECTS OF FIVE MODIFICATIONS

Although the five modifications to the Galerkin finite-element formulation (GFEF) are theoretically valid and useful, the degrees of improvement to numerical solution need to be tested

and compared with that of GFEF. For this purpose, a FORTRAN computer program was developed based on the entire formulation derived in this study. The program uses the linear hexahedron element only. The investigation is limited to one- and two-dimensional groundwater flows.

Verification Example for One-Dimensional Solute Transport

Analytical solutions for a one-dimensional advection-dispersion process have been extensively employed to verify numerical models (Sun and Yeh 1983; Wang et al. 1986; Istok 1989). The governing equation for such a process and the boundary and initial conditions can be expressed as

$$\frac{\partial C}{\partial t} = D\frac{\partial^2 C}{\partial x^2} - v_x\frac{\partial C}{\partial x}; \qquad C(x, 0) = 0; \qquad C(0, t) = C_0; \qquad \frac{\partial C(\infty, t)}{\partial x} = 0 \qquad (51a\text{–}d)$$

The solution to this set of equations was obtained by Van Genuchten and Alves (1982) as

$$C(x, t) = \frac{C_0}{2}\operatorname{erfc}\left(\frac{x - v_x t}{\sqrt{4Dt}}\right) + \frac{C_0}{2}\exp\left[\frac{v_x x}{D}\right]\operatorname{erfc}\left(\frac{x + v_x t}{\sqrt{4Dt}}\right) \qquad (52)$$

where v_x = average pore velocity in the x direction; C_0 = solute concentration at the upstream boundary; D = dispersion coefficient; exp[ ] = natural exponential function; and erfc( ) = complementary error function. As a verification example, the following aquifer properties were used: aquifer length L = 100 (m); average pore velocity v_x = 1.0 (m/day); longitudinal dispersivity coefficient a_L = 10 (m); $D = a_L v_x$; medium porosity θ = 0.3; $C(0, t) = C_0$ = 10.0 (g/m^3); and $C(x, 0) = 0$. The finite mesh for this problem consisted of 20 elements and 84 nodes with Δx = 5 (m), $\Delta_y = \Delta_z$ = 1 (m), and is shown in Fig. 1. The numerical formulation developed in this study was employed with weight factors $w_1 = w_2$ = 0.5. The Peclet number for this example is 0.5. The computed results for t = 5, 10, and 20 days are presented in Fig. 2, which shows that the model yielded quite accurate solutions.

Comparison between Mixed and Consistent Formulations

Let a_L = 1 m, Δt = 0.5 day, and all other conditions in the verification example (1) be kept the same. The Peclet number for this case is 5.0. The consistent formulation was tested using different values of the weighting factor $w = w_1 = w_2$, and the modified formulation used the fixed value of $w_1 = w_2$ = 0.5. Fig. 3 shows the computed results. The mixed formulation not only improved the solution accuracy, but also eliminated the numerical oscillation problem that occurred for the consistent formulation. Although the consistent formulation yielded a relatively

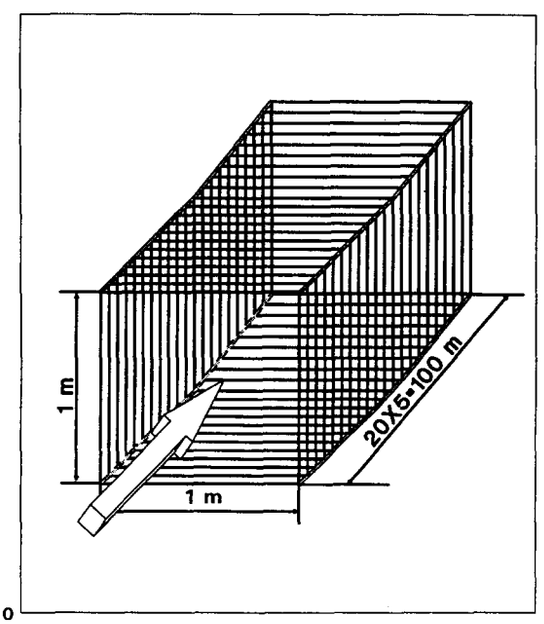

FIG. 1. 3D Mesh for 1D Flow Application

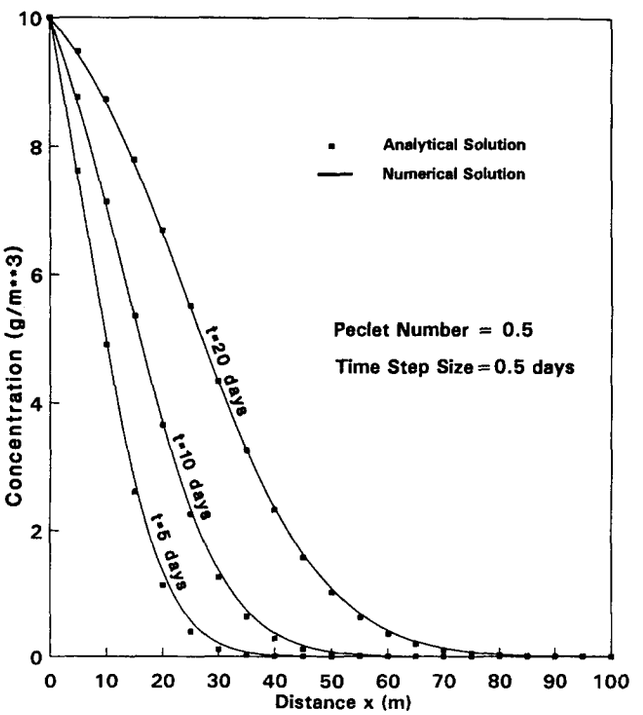

FIG. 2. Test of Overall Modified Formulations

J. Hydraul. Eng., 1995, 121(2): 145-158

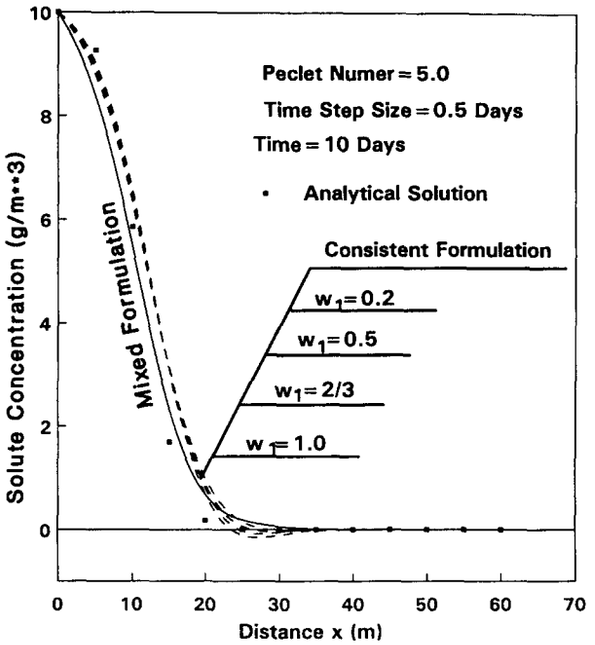

FIG. 3. Comparison of Two Formulations for Time-Derivative Term

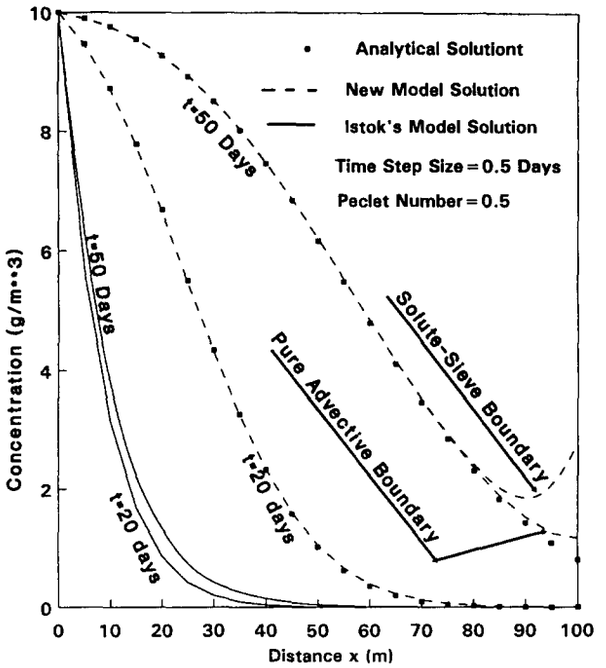

FIG. 4. Comparison of Two Formulations for Advection and Dispersion Terms

more accurate solution for $w_1 = 1.0$ as compared with $w_1 < 1.0$, this is not the case always. When the solution concentration in a flow field is rising, a smaller w_1 value will yield a more accurate solution. However, when $w_1 \leq 0.5$, the consistent formulation will usually yield oscillating or instable solutions.

Comparison of Modified and Conventional Formulation for Advection and Dispersion Terms

The conventional formulation programmed by Istok (1989) for computing the advection and dispersion terms was compared with the modified formulation proposed in this study. All conditions for the verification example (1) were kept the same. With $P_e = 0.5$ and $w_1 = w_2 = 0.5$, the computed results for $t = 20$ and 50 days are shown in Fig. 4. The boundary condition at the downstream end ($x = 100$ m) for the modified formulation was specified by $\{q\}\circ\{n''\} - \theta[D_m]\nabla C\circ\{n''\} \approx \{q\}\circ\{n''\}C \approx \bar{Q}_s C$, in which $\bar{Q}_s$ = average discharge rate at the downstream boundary. This average discharge rate can always be calculated using the known or computed velocity value of the boundary element and can be equally divided by the number of nodes on the boundary surface. In this example each of the four nodes on the downstream boundary surface has a discharge rate $\bar{Q}_s/4 = v_x\theta\Delta y\Delta z/4 = (1)(0.3)(1)(1)/4 = 0.075$ (m^3/s). These four nodal source values can be specified through the source matrix $[D_q]$. Fig. 4 clearly shows that the results from Istok's (1989) model are unacceptable and that the modified formulation yields very accurate results. To further demonstrate the usefulness of the modified formulation on the advection-dominated boundary, let us assume a special sieve or membrane is set up at the downstream boundary such that water can flow out of the boundary but the solute cannot. For this situation, the conventional Galerkin formulation certainly cannot handle this type of boundary. This is because when the conventional Galerkin formulation is used, advection fluxes are presumed to be automatically translated through boundaries and cannot be specified. On the other hand, the modified formulation automatically takes care of it. This is because all boundaries are considered solute impervious unless one specifies them, which is quite similar to modeling ground-water flow. The computed result for this solute-sieve boundary is shown in Fig. 4, which clearly shows the accumulation of solute at the downstream boundary.

Comparison between FISS and FDSS

To compare the two solution schemes we changed $a_1 = 1$ m, so that the Peclet number equals five. Four values of the weight factor ($w_1 = 0.2, 0.5, 2/3, 1$) were evaluated for the FDSS, keeping all other conditions the same as those in the verification example (1). All the modified formulations were used for the computation. When $w_1 = 0.5$ the two solutions became the same. However, the FDSS could choose any values of w_1 within the interval of [0, 1]. The

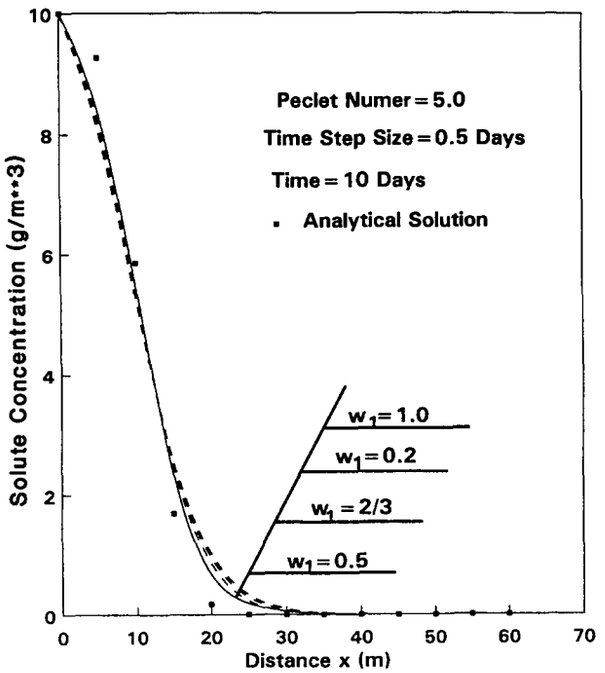

FIG. 5. Comparison of FISS with FDSS

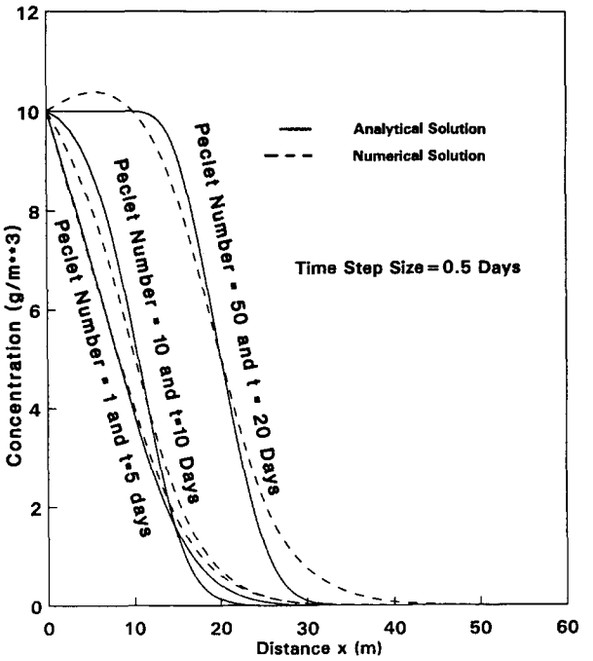

FIG. 6. Effect of Peclet Number on Model Solution

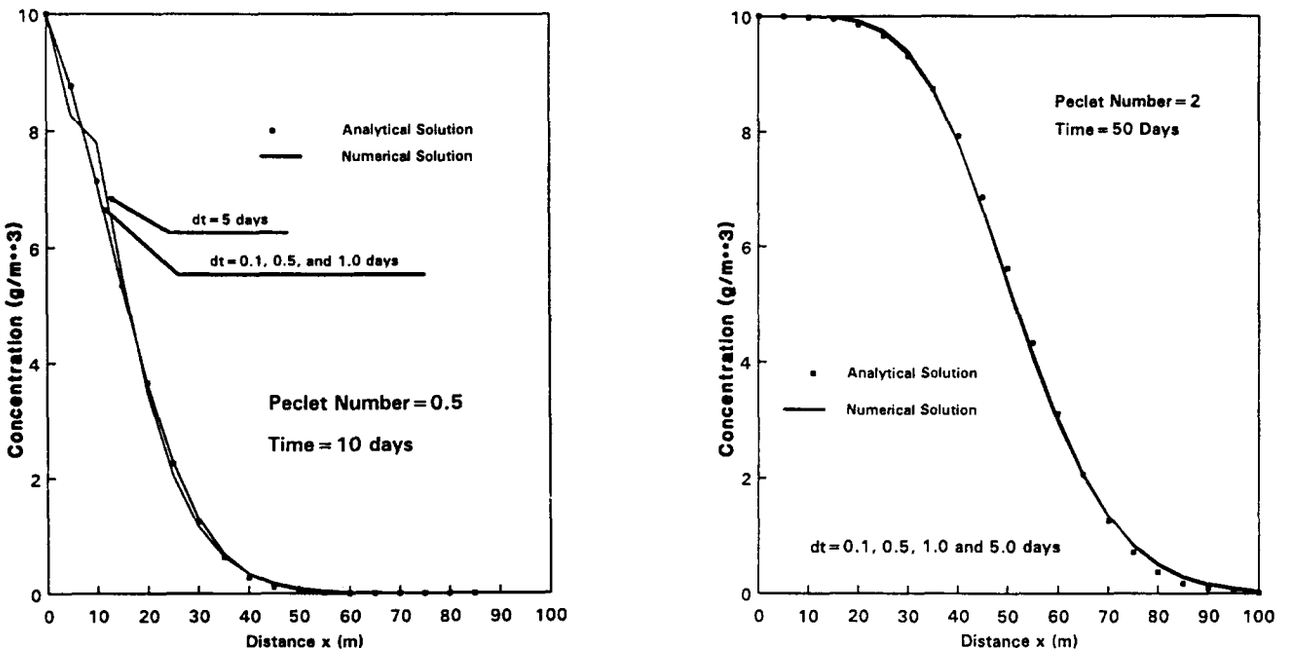

FIG. 7. Effect of Time Step Size on Model Solution

computed results at time $t = 10$ days are shown in Fig. 5. Clearly $w_1 = 0.5$ was the best choice. The use of other values of w_1 not only decreases the model solution accuracy, but also causes minor numerical oscillation. Fig. 5 also shows that $w_1 = 1.0$ was the worst choice for this example when the FDSS was used.

Effect of Peclet Number on Model Solution

To evaluate the effect of the five modifications on handling numerical oscillation and dispersion problems, all conditions given for the verification example (1) are kept the same except for changing the value of a_L to 5, 0.5, and 0.1, respectively, to make the Peclet number change to $P_e = 1$, 10, and 50, respectively. The computed results for $t = 5$ days in the first case, $t = 10$ days in the second case, and $t = 20$ days in the third case are shown in Fig. 6. This test example shows the model yielded a very accurate solution for $P_e = 1$; a reasonable solution for $P_e =$

10; and a slightly oscillating solution for $P_e = 50$. Fortunately, most solute-transport problems in a ground-water flow system have a Peclet number of less than 50.

Effect of Time Step Size on Model Solution

To examine model stability, the time step size Δt was changed from 0.1, 0.5, and 1.0 to 5.0 days, respectively. Two Peclet number cases were considered, with $Pe = 0.5$ and 2. All other conditions in the verification example (1) were kept unchanged. The computed results for $Pe = 0.5$ at $t = 10$ days and for $Pe = 2$ at $t = 50$ days are shown in Fig. 7. All computed results showed the model was relatively insensitive to the time step size, even if Δt reaches 5 days.

Numerical Example of Contaminated Well Field

Let there be a confined aquifer (Istok 1989) comprising two types of media: one was sandy gravel (elements 1 and 4), and the other was silty sand (elements 2, 3, 5, and 6). The hydraulic conductivity for the sandy gravel was $k_{xx} = 30$ (m/day), $k_{yy} = 10$ (m/day), and all other components were zero. Dispersivity coefficients for longitudinal and transverse directions were $a_L = a_T = 10$ (m); other medium and fluid parameters for elements 1 and 4 were $\lambda = 0.001$ (1/day), $\rho_b = 1{,}100$ (kg/m^3), $k_d = 0.00001$ (m^3/kg), $\theta = 0.3$, and $D^* = 0.0001$ (m^2/day). The hydraulic conductivity for the silty sand was $k_{xx} = 10$ (m/day), $k_{yy} = 5$ (m/day), and all other components were zero. Dispersivity coefficients for longitudinal and transverse directions were $a_L = a_T = 5$ (m), and other parameters for the silty sand medium were $\lambda = 0.001$ (1/day),

TABLE 1. Computed Hydraulic Head for Well Field

Node number (1)	Hydraulic head (m) (2)	Node number (3)	Hydraulic head (m) (4)
1	125.000	13	125.000
2	124.947	14	124.947
3	124.819	15	124.819
4	124.655	16	124.655
5	125.000	17	125.000
6	124.900	18	124.900
7	124.530	19	124.530
8	124.651	20	124.650
9	125.000	21	125.000
10	124.951	22	124.951
11	124.808	23	124.808
12	124.650	24	124.650

TABLE 2. Computed Velocity Head for Well Field

Element number (1)	q_x (m/s) (2)	q_y (m/s) (3)	q_z (m/s) (4)
1	0.3938	0.0077	0.0000
2	0.4418	−0.1004	0.0000
3	0.0172	−0.1307	0.0000
4	0.4581	−0.0145	0.0000
5	0.4987	0.0682	0.0000
6	0.0430	0.1381	0.0000

TABLE 3. Computed Solute Concentration

Node number (1)	Computed Solute Concentration (g/m^3)		
	$t = 0.5$ days (2)	$t = 2.5$ days (3)	$t = 5$ days (4)
1	100.000	100.000	100.000
2	32.719	79.958	91.504
3	6.210	46.144	83.563
4	0.378	6.500	18.734
5	100.000	100.000	100.000
6	20.219	65.529	87.953
7	3.538	27.056	50.591
8	0.405	8.462	30.370
9	100.000	100.000	100.000
10	26.187	75.273	90.938
11	3.916	34.641	71.959
12	0.131	2.579	8.725
13	100.000	100.000	100.000
14	32.719	79.958	91.504
15	6.210	46.145	83.563
16	0.378	6.500	18.734
17	100.000	100.000	100.000
18	20.219	65.529	87.953
19	3.538	27.056	50.591
20	0.405	8.462	30.370
21	100.000	100.000	100.000
22	26.187	75.273	90.938
23	3.916	34.641	71.959
24	0.131	2.579	8.725

ρ_b = 1,200 (kg/m^3), k_d = 0.00001 (m^3/kg), θ = 0.3, and D^* = 0.0001 (m^2/day). All other boundaries were impervious with respect to both water and solute. For simplicity, the domain was divided into six elements with 24 nodes. On the left side of the domain was a river with a constant head h = 125 (m). A pumping well penetrated the confined aquifer through nodes 7 and 19, with the pumping rate $Q = -2.5$ (m^3/day) at each node. The depth of the aquifer was assumed to be unity. Initially, solute concentration within the well field was zero. When the pumping process reached a steady state at time $t = t_0$, the river was suddenly polluted with a constant solute concentration C = 100 (g/m^3). Time step size was fixed as Δt = 0.5 (day). The first step of the computation was to run the 3D ground-water model developed by Yu and Singh (1993), to obtain the hydraulic heads at each node and the apparent velocity components for every element. The computed results are shown in Tables 1 and 2. The solute concentration at each node was computed with these results. Table 3 lists the computed nodal concentration at time t = 0.5, 2.5, and 5 days, respectively. Even though the largest element size ratio $\Delta x/\Delta z$ = 6,333.3, the computed solute concentration distribution appeared to be reasonable and numerical problems were not encountered.

CONCLUSIONS

The following conclusions were drawn from this study.

Five major modifications to the conventional Galerkin finite-element formulation were made in this study: (1) A mixed formulation (combination of the Galerkin method and the collocation method) for the time-derivative term of the governing equation was developed, and is a consistent way to derive the lumped formulation; (2) a general and useful numerical formulation for both the advective and dispersive terms of the governing equation was developed so problems with prescribed advection-dominated boundaries can be correctly handled; (3) a simpler and practical expression was derived for any problems with leaky boundaries, by using the unit step function; (4) proper expressions for sources and sinks were derived and some confusion, which appeared in the literature, may be clarified; and (5) a finite-integration solution scheme was developed to solve the system of ordinary-differential equations, and the limitations of using the finite-difference scheme were presented.

The effects of the five numerical modifications were investigated, based on one- and two-dimensional solute-transport examples. The computed results showed: First, the modified formulation for the advection and dispersion terms of the governing equation provided a way to correctly deal with the prescribed advection-dominated boundaries. Numerical test showed the modified formulation greatly improved the model solution accuracy and could be used for any type of boundary conditions, including the rare solute-sieve boundaries (possibly used in laboratory experiments). On the other hand, by using the conventional Galerkin formulation, only the dispersion-dominated boundary conditions can be specified; second, the finite-integration solution scheme could theoretically provide an exact solution to the system of ordinary-differential equation if the time step size is properly selected. Tests showed the weighting factors $w_1 = w_2 = 0.5$ gave the best model performance; third, the mixed formulation offers a consistent way to derive the lumped formulation because no special weight function is used to handle the time-derivative term; and fourth, solutions from the modified formulations were relatively insensitive to the time step size even if Δt increased up to 5 days.

APPENDIX I. REFERENCES

Bear, J. (1972). *Dynamics of fluids in porous media*. Elsevier, New York, N.Y.

Bear, J. (1979). *Hydraulics of groundwater*. McGraw-Hill, New York, N.Y.

Bear, J., and Verruijt, A. (1987). *Modeling groundwater flow and pollution*. D. Reidel Publ. Co., Dordrecht, Holland, The Netherlands.

Huyakorn, P. S., Jones, B. G., and Anderson P. F. (1986). "Finite element algorithms for simulating tree-dimensional groundwater flow and solute transport in multilayer systems." *Water Resour. Res.*, 22(3), 361–374.

Istok, J. (1989). "Groundwater modeling by the finite element method." *Water Resour. Monograph 13*, Am. Geophysical Union, Washington, D.C.

Park, N. S., and Liggett, J. A. (1990). "Taylor-least-squares finite element for two-dimensional advection dominated advection-diffusion problems." *Int. J. Numerical Methods in Fluids*, 11(1), 21–38.

Park, N. S., and Liggett J. A. (1991). "Application of Taylor-least-squares finite element to three dimensional advection diffusion equation." *Int. J. Numerical Methods in Fluids*, Vol. 13, 759–733.

Pinder, G. F., and Gray, W. G. (1977). *Finite element simulation in surface and subsurface hydrology*. Academic Press, New York, N.Y.

Segerlind, L. J. (1984). *Applied finite element analysis*. John Wiley and Sons, Inc., New York, N.Y.

Sun, N-Z., and Yeh, W. W. (1983). "A proposed upstream weight numerical method for simulating pollutant transport in groundwater." *Water Resour. Res.*, 19(6), 1489–1500.

Thomson, N. R., Sykes, J. F., and Lennox, W. C. (1984). "A Lagrangian porous media mass transport model." *Water Resour. Res.*, 20(3), 391–399.

Van Genuchten, M. T., and Alves, W. J. (1982). "Analytical solutions of the one-dimensional convective-dispersive solute transport equation." *Tech. Bull. No. 1661*, Agric. Res. Service, U.S. Dept. of Agr., Washington, D.C.

Wang, C., Sun, N., and Yeh, W. W. (1986). "An upstream weight multiple-cell balance finite-element method for solving three-dimensional convection-dispersion equations." *Water Resour. Res.*, 22(11), 1575–1589.
Wang, H. F., and Anderson, M. P. (1982). *Introduction to groundwater modeling*. W. H. Freeman and Co., San Francisco, Calif.
Wylie, C. R., and Barrett, L. C. (1982). *Advanced engineering mathematics*, 5th Edition, McGraw-Hill, New York, N.Y.
Yeh, G. T. (1986). "An orthogonal-upstream finite element approach to modeling aquifer contaminant transport." *Water Resour. Res.*, 22(6), 952–964.
Yu, F. X. (1992). "Modeling three-dimensional groundwater flow and solute transport by the finite element method with parameter estimation," PhD Dissertation, Louisiana State Univ., Baton Rouge, La.
Yu, F. X., and Singh, V. P. (1993). "A general 3-D groundwater flow model by using the modified galerkin finite element method." *J. Irrig. and Drain. Engrg.*, ASCE, in press.

APPENDIX II. NOTATIONS

The following symbols are used in this paper:

$[A]$ = element sorption matrix;
C = solute concentration (ML^{-3});
C_i^* = known solute concentration at ith source point (ML^{-3});
D = solution domain or dispersion coefficient;
$[D_a]$ = element advection matrix;
$[D_d]$ = element dispersion matrix;
D_e = element domain;
$[D_f]$ = element matrix contributed by first-order biochemical reaction;
D_i = subdomain consisting of all elements attached to node i;
$[D_m]$ = combined mechanical and molecular diffusion coefficient tensor;
$[D_q]$ = element matrix contributed by all sink points within element e;
g_1 = known solute-concentration function over solution domain D at time $t = 0$;
g_2 = known solute-concentration function over boundary surface S at time t;
g_3 = known solute-flux function (positive for inflow) through boundary surface S;
K_d = distribution coefficient (L^3M^{-1});
L = differential operator;
m = number of elements in mesh;
$N_i^{(e)}$ = basis function for node i of element e;
n = number of nodes of element e;
n_i = number of elements attached to node i;
n_k = number of surfaces of element;
$\{n^o\}$ = unit outward vector normal to boundary surface S;
n_s = number of sources in study domain;
n_w = number of pumping wells or sinks in domain;
p = number of nodes in mesh;
$Q_i(t)$ = source rate of fluid at ith source point (L^3T^{-1});
$Q_j(t)$ = sink rate of fluid (L^3T^{-1}), negative in value, at jth sink point;
$\{q\}$ = apparent velocity vector (LT^{-1});
R = retardation factor defined as $R = 1 + \rho_b K_d/\theta$;
R_i = sum of weighted residuals at node i;
$R_i^{(e)}$ = weighted residual at node i contributed by element e;
S = solution domain surfaces;
$u(\ )$ = unit step function;
w_1, w_2 = two weight factors;
$w_i(x, y, z)$ = weight function for ith nodal point;
δ = Dirac delta function;
θ = medium porosity;
λ = first-order biochemical reaction coefficient (T^{-1});
ρ_b = bulk density of porous media (ML^{-3});
ϕ = field variable;
$\hat{\phi}_i(x, y, z, t)$ = interpolation function for ith element; and
ϕ_i = value of field variable for jth node at time t.